FORSCHUNGSZENTRUM JÜLICH GmbH
Zentralinstitut für Angewandte Mathematik
D-52425 Jülich, Tel. (02461) 61-6402

Interner Bericht

AIX Compiler Update

Klaus Wolkersdorfer

FZJ-ZAM-IB-2001-17

September 2001

(letzte Änderung: 06.09.2001)

Vortrag: AIX-Arbeitskreis, 7.9.2001, Dresden

AIX Compiler Update

Klaus Wolkersdorfer
(K.Wolkersdorfer@fz-juelich.de)

Forschungszentrum Jülich GmbH
Zentralinstitut für Angewandte Mathematik (ZAM)
Tel: +49-2461-61-6579

XL FORTRAN 7.1

- Current Level: 7.1.0.2 (incl. IY16228:
  Performance degradation with –qarch=pwr3 and –O3)
- Full Fortran 95 Standard (and OpenMP)
- No html documentation search facility (just pdf-search)
- options we have additionally in /etc/xlf.cfg:
  -qnolm,-qhalt=E,-qmaxmem=16384,-L/usr/local/lib

C for AIX V5

- Current Level: 5.0.2.0
- Language Level ANSI/ISO 1990 C (and OpenMP)
- After install run once: /usr/vac/bin/replaceCSET
- Tools: memdbg, XLDB enhancements, SDE
- options we have additionally in /etc/vac.cfg:
  -qnolm,-qmaxerr=8:s,-L/usr/local/lib
- AIX 5.1: vac.cfg ➔ vac.cfg.43 instead vac.cfg.51

Visual Age C++ for AIX V5

- Current Level: 5.0.2.1
- Language Level ANSI/ISO 1998 C++
- Incremental Compiler + Batch Compiler (Makefile) !!!
- 64 bit support + Power 3-II Optimization !!!
- Tools: Visual Builder, Data Access Builder, LPEX Editor

Visual Age C++ for AIX V5 (cont.)

- xlC (and friends): /etc/vac.cfg **and** /etc/vacpp.cfg
- vacide: Integrated Development Environment
  - Insert /usr/vacpp/bin into PATH variable
  - Slow
  - Needs LUM (just information message)
  - AIX 5.1 with 64 bit kernel: IOT/Abort Trap

Compiler : 64 bit support (-q64)

- Good Doc: *AIX 64-bit Performance in Focus* (SG24-5103)
- LP64 model: 4/8/8 (Integer/Long/Pointer)
- Recommended option: -qwarn64
- Recommended option for xlf: –qintsize=8
- -bmaxdata, -bmaxstack necessary only if soft limits are set
- Every user can access virtually 16 millions TB = 16EB

Administration of Large Programs

- We wanted to limit *ordinary* users to 256MB
- Allow larger programs in off peak batch jobs
- ulimit –Hd 262144
  ➔ No compiler can run anymore (still in AIX 5L)
- All compilers require sofar unlimited data limit
- PMR 033,724,46368 open

LUM: License Use Management

- Choice between old NCS and direct binding
- We prefer direct binding only (efficiency reasons)
- We use 2 license servers
- One is acting as Central Registry also
- Distribute your licenses to both
  ➔ than they can immediately replace each other

LUM: Client Setup

- Install ifor_ls.base* and ifor_ls.client*
- Run on one client only: i4cfg -script
- Distribute /var/ifor/i4ls.ini to all clients
- Pitfall: Do not try to run i4blt on clients
  ➔ **Only** a copyright message will appear

LUM-Example: Enroll VAC++

- At Registry server:
  i4blt –a –f vacpp_c.lic –R kw –T 1
- Update to 99: (Customer controlled usage product)
  i4blt –U –v "'IBM Software Solutions Toronto'"
  -p "'VisualAge C++ Professional ' '5.0.c'" –T 99
- Distribute 49 of 99 to second license server 'srv':
  i4blt –E –v … -p … -A 49 –w srv

Remote Documentation Server

- For AIX: Standard
- For compiler: *AIX 5L Porting Guide* (SG24-6034) 4.11.2
- It works !!!
- Seems to be a problem with IMNSearch…2.1.3.0:
  To test this problem: search for 'longlong' in VAC doc:
  No hits with IMNSearch 2.1.3.0
  3 hits with IMNSearch 1.2.3.1

Tuning: Disk I/O Pacing

- Problem: One I/O, i.e. *cp* could take several minutes without letting somebody (i.e. compile, vi) in between
- Very important for multiuser systems
- Rule of 17 and 4:
  chsys –l sys0 –a maxpout=17 –a minpout=4
- Could also be done via *smitty chgsys*
- Prohibits one large I/O to monopolize system

Chisholm's Second Law:

'Anytime things appear to be going better…

…you sure have overlooked something'